BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 58, No. 1, 2010
*DOI: 10.2478/v10175-010-0013-7*

# Power equalization of AES FPGA implementation

M. STRACHACKI* and S. SZCZEPAŃSKI

Department of Microelectronic Systems, Gdansk University of Technology, 11/12 Narutowicza St., 80-952 Gdansk, Poland

**Abstract.** This paper briefly introduces side channel attacks on cryptographic hardware with special emphasis on differential power analysis (DPA). Based on existing countermeasures against DPA, design method combining power equalization for synchronous and combinatorial circuits has been proposed. AES algorithm has been implemented in Xilinx Spartan II-E field programmable gate array (FPGA) device using the standard and power-equalized methods. Power traces for DPA have been collected using XPower tool. Simulation results show that standard AES implementation can be broken after N=500 encryptions, while power-equalized counterpart shows no correlation between power consumption and the cipher key after N=2000 encryptions.

**Key words:** AES, FPGA, cryptography, side channel attack, DPA, power analysis, power equalization.

## 1. Introduction

To increase the level of security and prevent algorithm modification and cipher key readout, encryption algorithms are often implemented using an application specific integrated circuit (ASIC) or a field programmable gate array (FPGA). However underlying hardware can be attacked by [1]:

- fault injection (active invasive attack),
- tampering with clock signal and power supply (active non-invasive attack),
- side-channel information analysis: power consumption, electromagnetic radiation and processing time (passive attack).

These attacks were made in the past using inexpensive hardware [1–2]. To make these attacks difficult some countermeasures have been developed. This paper describes power equalization design method for synchronous and combinatorial circuits which enhances resistance against power analysis. Results of power analysis using XPower simulator for standard and power-equalized FPGA implementation of AES algorithm are presented.

## 2. Power analysis

The attack analyses power consumption during performing operations by underlying hardware and was introduced in [2].

Simple power analysis (SPA) is the immediate correlation between algorithm operation and power consumption. If an execution path of the algorithm depends on processed data, then SPA detects the sequence of operation and obtains information on the cipher key. The example is an attack on DES algorithm where difference in power consumption is observed for permutation, rotation and comparison instructions, especially in software [1–3].

Differential power analysis (DPA) is the immediate correlation between cipher key value and power consumption. DPA runs in two stages: first power consumption profile is collected for every encryption operation. Next correlation analysis is performed [4-5] or noise is filtered out using a distance of the mean test [2, 3, 6]. Both tests verify hypotheses on particular bytes of the cipher key.

In correlation analysis hypothetical key is used in encryption and the correlation between the power consumption for the unknown and hypothetical keys is computed. If a byte of the cipher key was guessed correctly, correlation reaches its maximum. DPA correlation attack on 8 most significant bits of the cipher key in AES algorithm is described in [4, 5].

In the distance of the mean test the average power consumption value is computed and then encryption using hypothetical key is performed. Power consumption values are assigned to one of two subsets by the selection function. If there is a correlation between mean power consumption of the whole set and one of the subsets, then a given subset determines the value of one bit [3, 6]. DPA distance of the mean attack on DES is described in [2, 3].

Inferential Power Analysis (IPA) consists of two stages: profiling and key extraction [7, 8]. During profiling stage statistical operations are performed on a large number of power traces to learn details of the implementation to find key operations and to identify key bits. During key extraction stage, the key is obtained from a very few power traces.

## 3. Countermeasures against DPA

Power analysis countermeasures base on its decorrelation with input data and key data or hiding its variations to increase the complexity of an attack [5]. Basic methods:

- power equalization by complementary logic introduction (known as balancing in [1],
- power decorrelation by additional operation introduction (e.g. masking),
- power randomization by additional random power consumption introduction.

*e-mail: marek.strachacki@intel.com

Power equalization balances combinatorial logic and register switching, making increased total power consumption key independent, which renders DPA impossible. Special libraries ensure power equalization on gate or transistor levels [5, 9].

Power decorrelation masks linear operations and modifies non-linear ones [10] requiring considerable algorithm changes and slowing it down [5]. SPA reveals only the Hamming weight of masked value. However it has been shown [11] that additive mask can easily be broken using Hamming attack.

Power randomization generates pseudorandom digital noise and requires additional computation, increasing power consumption or introducing the additional delay [5, 12]. This does not prevent DPA attack, but makes it inefficient due to necessity of data collecting in a very long period [3, 12].

## 4. Proposed design method

This paper describes power equalization method, which enhances DPA resistance of hardware implementation of encryption algorithm [13].

To equalize power consumption in synchronous circuits, the same number of flip-flops should switch on each clock cycle. Additional flip-flop $Q_2$ is added in parallel to the original flip-flop $Q_1$, and exactly one of them switches on each clock cycle:

$$Q_1(t-1) \oplus Q_1(t) \oplus Q_2(t-1) \oplus Q_2(t) = \text{‘1’u} \quad \text{oxp.} \qquad (1)$$

This method is proposed independently of [6].

To equalize power consumption combinatorial logic, the same number of outputs should switch on every single input change. Complementary logic function $g(x)$ is added in parallel to the original logic function $f(x)$, and exactly one output switches on each single input change:

$$f(x) \oplus g(x) = XOR(x). \qquad (2)$$

Function $f(x)$ can be of any type, which is a generalization of methods proposed in [9].

Joint application of both methods leads to the schematic shown in Fig. 1. Since it is impossible to insert additional blocks in the behavioural source code, the circuit is modified after logic synthesis.

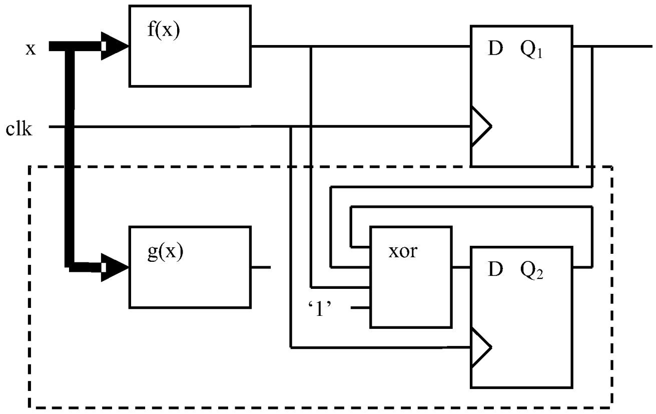

Fig. 1. Proposed method of power equalization. Additional elements are outlined by dotted line

## 5. Design requirements

Power consumption is most often estimated by counting flip-flop switching directly in VHDL code [4, 6] or measured in a real environment [3, 4]. In this paper we use XPower tool, which enables timed simulation of power consumption, used on early stage of design cycle, based on input value changed dump (VCD) file. However, XPower was oriented towards low power design worst case analysis, not to measure accurate power values. As a result, power values are overestimated, which does not matter as long as overestimation is proportional.

Hardware implementation was done in Xilinx Spartan-II E FPGA. Two AES [14] implementations were done: standard (without power equalization) and power-equalized, where complementary blocks had been added after logical synthesis. Iterative cipher architecture was chosen to simplify DPA. One AES round was implemented with internal pipeline registers after each operation. Encryption of 128-bit block with 128-bit key requires 10 iterations (40 clock cycles).

DPA was performed using correlation analysis as described in [4, 5] using random plaintexts for only N=2000 encryption operations due to XPower limitation [13]. During simulation round keys were generated and for each of N=2000 operations encryption module was reset and first four cycles (first encryption round) were performed, as first round subkey is equal to cipher key. Data generated by XPower was collected and then correlated with software-counted number of flip-flop switching for a hypothetical key.

## 6. Experiments and results

Two versions of AES cipher were implemented. Standard version uses 3931 configurable logic block (CLB) slices and minimal clock period is 15.5 ns. Power-equalized version requires 6831 CLB slices and minimal clock period is 18 ns. Almost twofold higher resource usage is caused by complementary logic generation for encryption module (for key expansion module complementary logic was not applied). Lower maximal frequency is the effect of more reprogrammable FPGA interconnections.

XPower generates power consumption data with resolution set by user. Too low resolution cumulates power values of several operations to one timeslot. Too high resolution disperses power values of one operation to several timeslots. After experiments resolution was set to flip-flop delay (about 1 ns), what ensures that power consumption connected with flip-flop switching is collected in one timeslot.

To extract the correct data from power traces, precise points of time should be determined for all operations. Two kinds of power simulations were carried out. The first one was performed after mapping process, taking flip-flop switching delay into account (1.2 ns delay) and the second one was performed after place and route process, additionally taking signal propagation delay into account (1.8 ns delay). The example of timing is shown in Fig. 2.

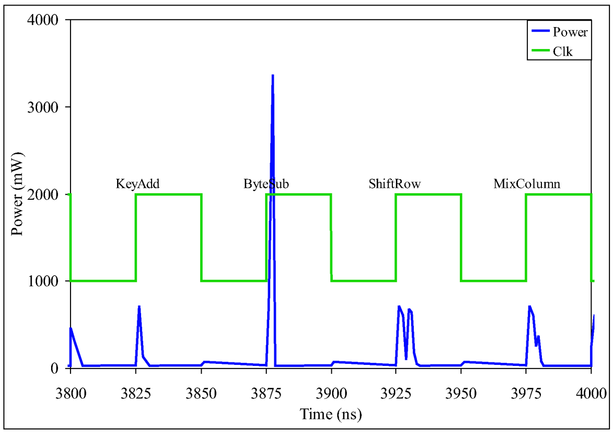

Fig. 2. Power consumption of standard implementation during the $1^{st}$ round of AES encryption

DPA was done for both AES implementations. DPA for the standard implementation showed that the cipher key can be determined by doing N=500 encryptions and analyzing power consumption of KeyAdd, ByteSub or ShiftRow operations in the first encryption round, as shown in Fig. 3, modified implementation, flip-flop switching power consumption was almost constant, oscillating within 5% range. DPA for N=2000 encryptions showed no correlation between the power consumption and a key value as shown in Fig. 5. This proves that proposed design method enhances DPA resistance.

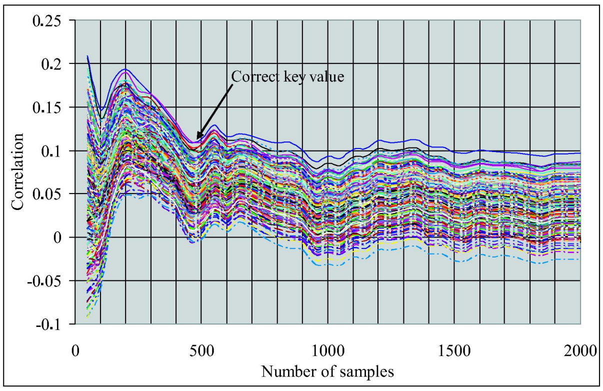

Fig. 3. Correlation for KeyAdd operation of standard implementation

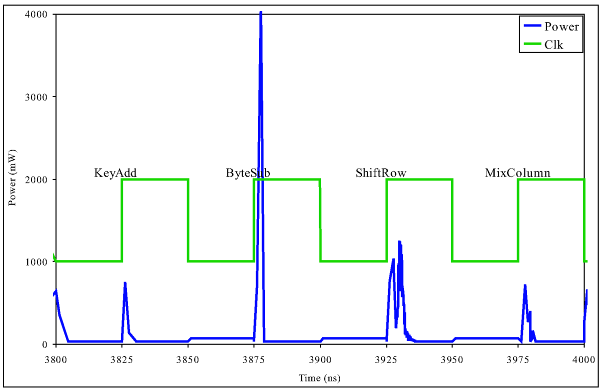

Fig. 4. Power consumption of modified implementation during the $1^{st}$ round of AES encryption

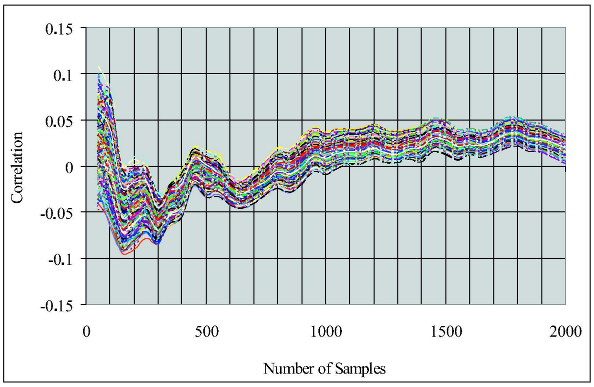

Fig. 5. Correlation for KeyAdd operation of modified implementation

## 7. Conclusions

It was shown that successful attack on AES algorithm can be done within N=500 encryption operations. Power equalized implementation method was proposed, considering synchronous circuits and combinatorial logic. Power consumption of both standard and modified AES implementations was simulated using XPower tool. Based on XPower generated data it was proven that the proposed design method effectively enhances DPA resistance.

REFERENCES

[1] H. Bar-El, “Introduction to side channel attacks”, in *White Paper*, Discretix Technologies Ltd, Israel, 1999.

[2] P. Kocher, J. Jaffe, and B. Jun, “Introduction to differential power analysis and related attacks”, *Technical Report, Cryptography Research Inc.* 1, http://www.cryptography.com/dpa/technical (1998).

[3] P. Kocher, J. Jaffe, and B. Jun, “Differential power analysis”, *Advances in Cryptology: Proc. CRYPTO-99* LNCS 1666, 388–397 (1999).

[4] S. Ors, F. Gurkaynak, E. Oswald, and B. Preneel, “Power-analysis attack on an ASIC AES implementation”, *Int. Conf. Information Technology: Coding and Computing* 1, 546–552 (2004).

[5] M. Pierson, B. Brady, *Low Cost Differential Power Analysis (DPA) Resistant Crypto-Chips,* University of California, California, 2006.

[6] L. McDaniel, *An Investigation of Differential Power Analysis on FPGA-Based Encryption Systems*, Virginia Polytechnic Institute, Virginia, 2003.

[7] P. Fahn and P. Pearson, “IPA: A new class of power attacks”, *Proc. 1st Int. Workshop on Cryptographic Hardware and Embedded Systems* 1717, 173–186 (1999).

[8] J. Gawinecki and P. Bora, “Safety analysis of imptementation of equipment algorithms of information coding”, *National Symposium of Telecommunication and Telecomputing* 1, CD-ROM (2006), in Polish.

[9] K. Tiri and I. Verbauwhede, “Synthesis of secure FPGA implementation”, *Int. Workshop on Logic and Synthesis* 1, 224–231 (2004).

[10] M. Akkar and C. Giraud, “An implementation of DES and AES, secure against some attacks”, *Int. Workshop on Cryptographic Hardware and Embedded Systems* 2162, 309–318 (2001).

[11] M. Gomułkiewicz and M. Kutyłowski, "Hamming weight attacks on cryptographic hardware – breaking masking defense", $7^{th}$ *Proc. Eur. Symposium on Research in Computer Security* 2502, 90–103 (2002).

[12] L. Benini, A. Macii, E. Macii, E. Omerbegovic, M. Poncino, and F. Pro, „Energy-aware design techniques for differential power analysis protection", *Design Automation Conf.* 1, 36–41 (2003).

[13] M. Strachacki and S. Szczepański, "Implementation of AES algorithm resistant to differential power analysis", *15th Proc. Int. Conf. Electronic, Circuits and Systems* 1, 214–217 (2008).

[14] J. Daemen and V. Rijmen, "AES proposal: rijndael", *Proc. First AES Candidate Conf.* 1, CD-ROM (1998).